View metadata, citation and similar papers at core.ac.uk
brought to you by CORE
provided by University of Queensland eSpace

A COMPUTATIONALLY EFFICIENT IMPLEMENTATION OF QUADRATIC TIME-FREQUENCY DISTRIBUTIONS

John M. O' Toole, Mostefa Mesbah and Boualem Boashash

Perinatal Research Centre, University of Queensland,
Royal Brisbane & Women's Hospital, Herston, QLD 4029, Australia.
e-mail: j.otoole@uq.edu.au

ABSTRACT

Time-frequency distributions (TFDs) are computationally intensive methods. A very common class of TFDs, namely quadratic TFDs, is obtained by time-frequency (TF) smoothing the Wigner Ville distribution (WVD). In this paper a computationally efficient implementation of this class of TFDs is presented. In order to avoid artifacts caused by circular convolution, linear convolution is applied in both the time and frequency directions. Four different kernel types are identified and separate optimised implementations are presented for each kernel type. The computational complexity is presented for the different kernel types.

1. INTRODUCTION

TFDs are two dimensional representations of signal energy in the joint TF domain [1]. One of the most used classes of TFDs is the class of quadratic shift-invariant TFDs (henceforth referred to as simply TFDs). The most basic TFD of this class is the WVD defined, for a real signal $s(t)$, as

$$W_z(t,f) = \int_{-\infty}^{\infty} z(t+\tfrac{\tau}{2}) z^*(t-\tfrac{\tau}{2}) \mathrm{e}^{-j2\pi\tau f} \mathrm{d}\tau$$

where $z(t)$ is the analytic associate of $s(t)$ [1, pp. 13]. The reason the analytic associate, $z(t)$, is used rather than real signal $s(t)$ is twofold: firstly it eliminates the crossterms between the positive and negative frequency components in the TF domain and secondly it makes an alias-free discrete representation possible [2]. The general form of this class of TFDs can be expressed using the TF kernel $\gamma(t,f)$ as follows:

$$\rho_z(t,f) = W_z(t,f) \underset{t}{*}\underset{f}{*} \gamma(t,f) \qquad (1)$$

where $*$ represents the convolution operation.

1.1. Discrete Time-Frequency Distributions

The implementation of TFDs for digital signal processing purposes requires a discrete version of the continuous TFD defined in (1). This requires that both the time and frequency arguments in $\rho_z(t,f)$ be discretised. As TFDs are obtained from mapping a time-domain signal $s(t)$, then likewise discrete TFDs (DTFDs) need to be obtained from the discrete-time signal $s(nT)$, where T represents the sampling period. A commonly used definition of the discrete WVD (DWVD) was proposed in [3], and for a signal $s(nT)$ can be represented as

$$W_x^{(C)}(nT, \tfrac{k}{2NT}) \overset{\text{def}}{=} 2T \sum_{m=0}^{N-1} K_x(nT, 2mT) \mathrm{e}^{-j\frac{2\pi}{N}mk} \qquad (2)$$

for $n = 0, 1, \ldots, N-1$ and $k = 0, 1, \ldots, N-1$. The temporal instantaneous autocorrelation function (TIAF) K_x is defined as

$$K_x(nT, 2mT) \overset{\text{def}}{=} x(T[n+m]_{2N}) x^*(T[n-m]_{2N}) \qquad (3)$$

where $[a]_N \overset{\text{def}}{=} a \mod N$. $x(nT)$ is defined as

$$x(nT) \overset{\text{def}}{=} \begin{cases} z(nT), & 0 \le n \le N-1 \\ 0, & N \le n \le 2N-1 \end{cases} \qquad (4)$$

where $z(nT)$ represents the discrete-time analytic associate of $s(nT)$, using the frequency domain method described in [4]. At this point it is worth noting that although $z(nT)$ has a completely zero negative spectrum (implicit from its definition [4]), $x(nT)$ does not. Also the original definition in [3] of (2) uses the TIAF $K_z(nT, 2mT) = z((n+m)T) z^*((n-m)T)$ where it is assumed that $z(nT)$ is zero outside the interval $[0, N-1]$. This results in the same distribution as (2).

The DWVD is convolved in both time and frequency directions with the discrete TF kernel $\gamma(nT, \frac{k}{2NT})$ to produce the DTFD $\rho_x^{(C)}(nT, \frac{k}{2NT})$. In order to avoid artifacts introduced from circular convolution (see, e.g., [5]), linear convolution can be applied in both the time and frequency directions. The linear convolution can be implemented efficiently using discrete Fourier transforms (DFTs) [5, pp.

576], which in turn is implemented by fast Fourier transform (FFT) algorithms. The DTFD can be expressed as

$$\rho_x^{(C)}(nT, \tfrac{k}{2NT}) = \underset{m\to k}{\mathrm{DFT}_L}\left\{\underset{l\to n}{\mathrm{IDFT}_L}\left\{A_x^{(C)}(\tfrac{l}{LT}, mT)g(\tfrac{l}{LT}, mT)\right\}\right\} \quad (5)$$

where DFT_L and IDFT_L denotes the DFT and inverse DFT of length L respectively. It is assumed in (5) that $L = 2N$, thus satisfying the linear convolution condition that $L > 2N-1$. The interpolated discrete ambiguity function (DAF) $A_x^{(C)}$ given by

$$A_x^{(C)}(\tfrac{l}{LT}, mT) = \underset{n\to l}{\mathrm{DFT}_L}\left\{\underset{k\to m}{\mathrm{IDFT}_L}\left\{W_x^{(C)}(nT, \tfrac{k}{2NT})\right\}\right\} \quad (6)$$

is obtained from the zero-padded (in both the time and frequency direction) $L \times L$ DWVD. The kernel g can be similarly defined from γ. Assuming a closed form expression for the kernel g exists, then the computational load required to obtain the DTFD is mainly dependent on the construction of the DAF in (6). It is obvious that computing the DAF from the zero-padded DWVD as described in (6) is not a very efficient approach, as a DAF can be computed directly from a TIAF, by applying a DFT on the TIAF as $n \to l$.

It is worth noting that $A_{\hat{x}}^{(C)}(\frac{l}{LT}, mT) \neq A_x^{(C)}(\frac{l}{LT}, mT)$, with $\hat{x}(nT/2)$ defined as

$$\hat{x}(\tfrac{nT}{2}) \stackrel{\mathrm{def}}{=} \begin{cases} \hat{z}(\frac{nT}{2}), & 0 \le n \le 2N-1 \\ 0, & 2N \le n \le 4N-1 \end{cases}$$

where $\hat{z}(nT/2)$ is formed by interpolating $z(nT)$. As $\hat{x}$ now has half the bandwidth of x, therefore $W_{\hat{x}}^{(C)}$ should be zero for half of the spectral range. However, as $\hat{x}$ is not exactly zero for negative frequencies, then $W_{\hat{x}}^{(C)}$ will not be exactly zero for half of the spectral range, due to aliasing in the frequency direction [2].

A computationally efficient implementation of DTFDs using TF linear convolution will now be presented. This method is more efficient than the one in (6) for the computation of a DAF.

2. A MODIFIED DWVD

In an effort to produce an efficient implementation for DTFDs, a modified DWVD will be presented in this section. This distribution is simply defined as

$$W_y^{(M)}(nT, \tfrac{k}{2NT}) \stackrel{\mathrm{def}}{=} 2T\sum_{m=0}^{N-1} K_y^{(M)}(nT, 2mT)\mathrm{e}^{-j\frac{2\pi}{N}mk} \quad (7)$$

with

$$K_y^{(M)}(nT, 2mT) \stackrel{\mathrm{def}}{=} K_y(nT, 2mT) + K_y(nT, 2(2N-m)T)$$

where $K_y(nT, 2mT) = y(T[n+m]_{2N})y^*(T[n-m]_{2N})$ and $y(nT)$ is the analytic associate of $\Re\{x(nT)\}$ for $n \in \{0, 1, \ldots, 2N-1\}$, with $x(nT)$ defined in (4). The defined TIAF $K_y^{(M)}$ is of dimension $N \times N$, as K_y, which is of dimension $N \times 2N$, is folded in the lag direction. Thus $\underset{m\to k}{\mathrm{DFT}_N}\left\{K_y^{(M)}(nT, 2mT)\right\}$ results in a DWVD with a frequency domain sampling rate of $k/(2NT)$, as described in (7).

Although this modified DWVD was proposed for the purpose of enabling efficient linear convolution, as will be explained in the next section, the distribution also has the advantage of being completely alias free in the frequency direction. This is due to the fact that the signal used in the distribution, namely $y(nT)$, has a completely zero negative spectrum, which is the assumption used for the creation of an alias-free DWVD [2]. This is in contrast to the conventional DWVD definition, $W_x^{(C)}$, as $x(nT)$ has some energy in the negative spectrum, as mentioned in the previous section. Although this aliasing may be smoothed out by the TF kernel for the DTFD $\rho_x^{(C)}$, the degree to which this aliasing is suppressed is dependent on the nature of the TF kernel.

To illustrate the aliasing effect of this negative spectral energy on $W_x^{(C)}$, this distribution was compared with the modified distribution $W_y^{(M)}$. For this example, an analytic associate, $z(nT)$, of a linear frequency modulated (LFM) signal of the form $s(nT) = \cos(2\pi nT(0.1 + (0.15/N)nT))$ for $n = 0, 1, \ldots, N-1$, with $T = 1$ and $N = 128$ was used to compute the DWVDs. The DWVDs were obtained from $z(nT)$ according to their definitions in (2) and (7). This aliasing effect is most notably observed in the parts of the distribution where no significant signal energy is expected, as the energy in the negative spectrum of $x(nT)$ is relatively small. Thus in Fig. 1, $W_x^{(C)}$ appears to have more ripple away from the signal component than $W_y^{(M)}$.

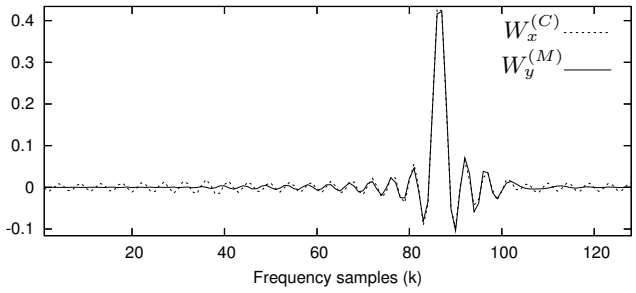

Figure 1: A comparison of the two DWVDs, $W_x^{(C)}(nT, \frac{k}{2NT})$ and $W_y^{(M)}(nT, \frac{k}{2NT})$ taken at the time slice $n = 100$.

3. EFFICIENT FORMATION OF DTFDS

Using the proposed DWVD, an interpolated DAF can be expressed as

$$A_y^{(M)}(\tfrac{l}{LT}, mT) \stackrel{\text{def}}{=} \underset{n\to l}{\text{DFT}_L}\left\{\underset{k\to m}{\text{IDFT}_L}\left\{W_y^{(M)}(nT, \tfrac{k}{2NT})\right\}\right\}. \quad (8)$$

This interpolated DAF can also be found directly from a TIAF if signal $\hat{y}(nT/2)$ is used, with

$$A_{\hat{y}}^{(M)}(\tfrac{l}{LT}, mT) \stackrel{\text{def}}{=} \underset{n\to l}{\text{DFT}_L}\left\{K_{\hat{y}}^{(M)}(nT, mT)\right\} \quad (9)$$

where signal $\hat{y}(nT/2)$ is obtained by interpolating $y(nT)$ using the DFT method [4]. This interpolated signal $\hat{y}$ is still an analytical signal in the sense that $\Re\{\hat{y}(nT)\} = s(nT)$, for $n \in \{0, 1, \ldots, N-1\}$, it has a zero negative spectrum, and satisfies the orthogonality property as discussed in [4]. As $\hat{y}(nT/2)$ has exactly zero negative spectral energy, then the DWVD formed from this signal, namely $W_{\hat{y}}^{(M)}$, is exactly zero for half of the frequency range of the distribution. Therefore $A_{\hat{y}}^{(M)}(\frac{l}{LT}, mT) = A_y^{(M)}(\frac{l}{LT}, mT)$, where $A_{\hat{y}}^{(M)}$ is obtained from the TIAF of signal $\hat{y}(nT/2)$, as in (9), and $A_y^{(M)}$ is formed from the DWVD of signal $y(nT)$, as in (8).

The DTFD can be computed efficiently using this DAF, $A_{\hat{y}}^{(M)}$, as similar to (5), this distribution can be expressed as

$$\rho_{\hat{y}}^{(M)}(nT, \tfrac{k}{2NT}) = \underset{m\to k}{\text{DFT}_L}\left\{\underset{l\to n}{\text{IDFT}_L}\left\{A_{\hat{y}}^{(M)}(\tfrac{l}{LT}, mT)g(\tfrac{l}{LT}, mT)\right\}\right\} \quad (10)$$

for $n = 0, 1, \ldots, N-1$ and $k = 0, 1, \ldots, N-1$. The kernel $g(\frac{l}{LT}, mT)$ in (10) is in its most general form, which is known as the nonseparable kernel. In an effort to further reduce the computation load, three other specific types of kernels will be defined and a particular implementation described for each kernel type. The three kernel types are: separable, doppler-independent (DI), and lag-independent (LI).

3.1. Separable Kernel

For the separable kernel, the kernel consists of two independent functions, namely $g(\frac{l}{LT}, mT) = G_1(\frac{l}{LT})g_2(mT)$, where g_1 (IDFT of G_1) and g_2 are of length Q and P respectively. A slightly different approach than the one used in (10) is applied with this kernel type, in an effort to minimise the computational load. Firstly a pseudo-DWVD is formed as

$$W_{\hat{y}}^{(M)}(nT, \tfrac{k}{2PT}; g_2) \stackrel{\text{def}}{=} \underset{m\to k}{\text{DFT}_{2P}}\left\{K_{\hat{y}}^{(M)}(nT, mT)g_2(mT)\right\} \quad (11)$$

for $k = 0, 1, \ldots, P-1$. The DTFD is then calculated as

$$\rho_{\hat{y}}^{(M)}(nT, \tfrac{k}{2PT}) = \underset{l\to n}{\text{IDFT}_R}\left\{\underset{n\to l}{\text{DFT}_R}\left\{W_{\hat{y}}^{(M)}(nT, \tfrac{k}{2PT}; g_2)\right\}G_1(\tfrac{l}{RT})\right\}.$$

with $R \stackrel{\text{def}}{=} N + Q - 1$ for $n = 0, 1, \ldots, N-1$.

3.2. Doppler-Independent Kernel

The DTFD when using the DI kernel, given by $g(\frac{l}{LT}, mT) = g_2(mT)$, is obtained from the pseudo-DWVD described in (11).

3.3. Lag-Independent Kernel

The DTFD for the LI kernel, given by $g(\frac{l}{LT}, mT) = G_1(\frac{l}{LT})$, can be efficiently realised as

$$\rho_{\breve{y}}^{(M_1)}(nT, \tfrac{k}{2NT}) = \underset{l\to n}{\text{IDFT}_L}\left\{\mathcal{K}_{\breve{y}}^{(M_1)}(\tfrac{l}{LT}, \tfrac{k}{2NT})G_1(\tfrac{l}{LT})\right\} \quad (12)$$

for $n = 0, 1, \ldots, N-1$ where the spectral instantaneous autocorrelation function (SIAF) is defined as

$$\mathcal{K}_{\breve{y}}^{(M_1)}(\tfrac{l}{LT}, \tfrac{k}{2NT}) \stackrel{\text{def}}{=} \breve{Y}_e(\tfrac{1}{LT}[2k+l]_L)\breve{Y}_e^*(\tfrac{1}{LT}[2k-l]_L) + \breve{Y}_o(\tfrac{1}{LT}[2k+l]_L)\breve{Y}_o^*(\tfrac{1}{LT}[2k-l]_L)\mathrm{e}^{-j\frac{2\pi}{Q}l}$$

with $L = 2N$ and where

$$\breve{Y}_e(\tfrac{l}{LT}) \stackrel{\text{def}}{=} \underset{n\to l}{\text{DFT}_L}\{\breve{y}(2nT)\}$$
$$\breve{Y}_o(\tfrac{l}{LT}) \stackrel{\text{def}}{=} \underset{n\to l}{\text{DFT}_L}\{\breve{y}[(2n+1)T]\}.$$

The signal $\breve{y}$ is defined as

$$\breve{y}(nT) \stackrel{\text{def}}{=} \begin{cases} y(nT), & 0 \le n \le 1N-1 \\ 0, & N \le n \le 2N-1. \end{cases}$$

The reason why this SIAF consists of separate DFTs of the signal's odd samples, $\breve{Y}_o$, and even samples, $\breve{Y}_e$, is due to the fact that the quadratic signal representation for $W_y^{(M)}$, i.e. some form including the terms $y(aT)y^*(bT)$, only exists for a, b both even or both odd [6]. The DTFD $\rho_{\breve{y}}^{(M_1)}$ only approximates the DTFD $\rho_y^{(M)}$ defined by

$$\rho_y^{(M)}(nT, \tfrac{k}{2NT}) = \underset{l\to n}{\text{IDFT}_L}\left\{\underset{n\to l}{\text{DFT}_L}\left\{W_y^{(M)}(nT, \tfrac{k}{2NT})\right\}G_1(\tfrac{l}{LT})\right\} \quad (13)$$

with about a maximum of a 2% (relative to the peak of the DTFD) error between the distributions. Although (13) provides the exact solution, (12) provides a more efficient implementation. The discrepancy between the distributions is due to the fact that $y(nT)$ is defined as having an exactly zero negative spectrum, where $\Im\{y(nT)\} \neq 0$ for $n = N, N+1, \ldots, 2N-1$. The requirement for linear convolution in the time direction is that the signal used for the SIAF needs to be exactly zero for the time period $n = N, N+1, \ldots, 2N-1$. Forcing $y(nT)$ to zero in this time period results in $\breve{y}(nT)$. This creates a comprise between y and x, as empirical findings revealed that $\breve{y}$ has less negative spectral energy than x. Exact linear convolution can be preformed efficiently with some aliasing in the frequency direction of the DTFD.

The notation $\rho_{\hat{y}}^{(M*)}$ will henceforth be used to collectively represent $\rho_{\hat{y}}^{(M)}$ as the DTFD with either the nonseparable, separable, or DI kernel, and $\rho_{\hat{y}}^{(M_1)}$ as the DTFD with the LI kernel type.

4. DISCUSSION

Each kernel type can be assessed in terms of computational load. As this load is dominated by the number and type of DFTs used by the distribution, these transforms can be easily counted and totalled. It is assumed that all terms N, R, P are of the form 2^a, with $a \in \mathbb{Z}$, and that the FFT method used to implement DFT_N (or IDFT) takes $\mathcal{O}(cN \log_2 N)$ real multiplications plus additions. The c constant is determined by the specific FFT algorithm used. When the DFT (or IDFT) is operating on conjugate symmetrical data, such as on $K_{\hat{y}}^{(M)}$ as $m \to k$, the DFT can be efficiently implemented using a commonly available real FFT method, as detailed in [6], which equates to a load of $\mathcal{O}(cN/2 \log_2 N)$.

Kernel Type	$\rho_x^{(C)}$	$\rho_{\hat{y}}^{(M*)}$
nonseparable	$\mathcal{O}(c10N^2 \log_2 2N)$	$\mathcal{O}(c5N^2 \log_2 2N)$
separable	$\mathcal{O}(c\frac{3}{2}PR \log_2 R$ $+cNP \log_2 2P$ $+c5N^2 \log_2 2N)$	$\mathcal{O}(c\frac{3}{2}PR \log_2 R$ $+cNP \log_2 2P)$
DI	$\mathcal{O}(cNP \log_2 2P$ $+cN^2 \log_2 2N)$	$\mathcal{O}(cNP \log_2 2P)$
LI	$\mathcal{O}(cN^2 \log_2 2N)$	$\mathcal{O}(cN^2 \log_2 2N)$

Table 1: Computational complexity for the conventional DTFD $\rho_x^{(C)}$ and the proposed DTFD $\rho_{\hat{y}}^{(M*)}$ each with different kernel types for signal of length N. P and R exist within the range $1 < P \leq N$ and $N < R \leq 2N-1$.

The computational complexity for the DTFDs with the different kernel types is presented in Table 1. The proposed DTFD $\rho_{\hat{y}}^{(M*)}$ is compared with the conventional DTFD $\rho_x^{(C)}$, where $\rho_x^{(C)}$ is computed using the expressions in (5) and (6). Similar kernel specific optimisations are applied to $\rho_x^{(C)}$. For the all the kernel types, except the LI kernel, the order of computational complexity for $\rho_{\hat{y}}^{(M*)}$ is approximately half that required for $\rho_x^{(C)}$.

5. CONCLUSION

A computational efficient implementation for quadratic TFDs was presented. The approach employs linear convolution of the DWVD with the TF kernel in both the time and frequency directions. To the best of the authors knowledge, no efficient implementation of this type has been previously presented. A modified DWVD was presented to enable this efficient implementation. This modified DWVD has also the advantage of being completely alias-free in the frequency direction, unlike the conventional definition. Different optimised implementations were presented for three special cases of the general nonseparable kernel. For the LI kernel an approximate realisation was given in order to reduce the computational load, while only slightly reducing the accuracy.

6. REFERENCES

[1] B. Boashash, Ed., *Time-Frequency Signal Analysis and Processing : A Comprehensive Reference*. Oxford, UK: Elsevier, 2003.

[2] B. Boashash, "Note on the use of the Wigner distribution for time-frequency signal analysis," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, no. 9, pp. 1518–1521, 1988.

[3] T. Claasen and W. Mecklenbraüker, "The Wigner distribution - a tool for time-frequency signal analysis - part 2: Discrete-time signals," *Philips J. Research*, vol. 35, pp. 276–350, 1980.

[4] S. L. Marple, Jr., "Computing the discrete-time 'analytic' signal via FFT," *IEEE Trans. Signal Processing*, vol. 47, no. 9, pp. 2600–2603, 1999.

[5] A. V. Oppenheim and R. W. Schafer, *Discrete-Time Signal Processing*. Englewood Cliffs, NJ 07458: Prentice-Hall, 1999.

[6] J. O' Toole, M. Mesbah, and B. Boashash, "A discrete time and frequency Wigner Ville distribution: Properties and implementation," in *Proc. Int. Conf. on Digital Signal Processing and Comm. Systems*, vol. CD-ROM, Dec. 19–21, 2005.